Identity-Based Cloud Authentication Protocol

Andrea Huszti, Norbert Oláh

Abstract: Cloud computing is a recently developed new technology for complex systems with massive-scale services sharing among many users. One of the most important security objectives is to achieve secure user authentication. If it is breached, confidentiality and integrity of the data may be compromised. We propose an identity-based mutual authentication protocol, where identity verification is carried out by multiple servers applying secret sharing technology on the cloud provider side. Robustness and elasticity of the cloud are ensured by its hierarchical structure.

Keywords: cloud computing, authentication, secret sharing, identity-based cryptography

Introduction

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. OpenStack is one of the most popular cloud computing software. OpenStack Identity service supports multiple methods of authentication, including user name and password, LDAP, and external authentication methods (i.e. Kerberos). There are fears about these systems due to their centralized structure. Services storing password information for a large number of enterprises in a central database are primary targets for hackers (e.g. Golden Ticket Attack [10,12], OneLogin attack). In the scientific literature generally centralized, one-factor [8,9] or two-factor identity verification protocols [3,4] are proposed. However, the concept of multiple-server model may enhance the security level. Brainard et.al. suggested a two-server approach in [2], where two servers together decide on the correctness of the password submitted for authentication. In [6], a multiple-server authentication protocol is designed, where one-time passwords are shared among the cloud servers. Merkle tree or a hash tree [11] is applied for verifying the correctness of the one-time password.

Boneh and Franklin [1] formalized the notion of Identity-Based Encryption (IBE) using bilinear pairings over elliptic curve groups. In IBE setting, the public key of a user can be any arbitrary string, typically the e-mail address. There is no need for Bob to go to the Certificate Authority to verify the public key of Alice. This way an IBE can greatly simplify certificate management. The authors presented an identity-based authentication for cloud computing in [7], based on the identity-based hierarchical model for cloud computing. There are several servers participating in the identity verification process, for each user authentication one encryption and one signature generation are needed.

We propose an identity-based mutual cloud authentication protocol, where user identity is verified by multiple servers. A user authenticates himself by his password with a help of a smart card capable of cryptographic calculations and storing values securely. For each authentication users should give their passwords, hence end-to-end identity verification is achieved. Users decide about the number of servers participate during authentication. The structure of the servers is hierarchical ensuring robustness and elasticity for the cloud. Due to the bilinear property we accomplish the multiple server key exchange and password registration with a single message transmission. In the authentication phase secret sharing is applied, mutual authentication is achieved only by two modular exponentiations besides the fast hash calculations and xor operations.

The proposed scheme

In this section, we describe our protocol. We differentiate two participants: *Users* (U) ask for services from the cloud service provider consisting of several *cloud servers* (C_i). A cloud server which is chosen randomly proceeds the steps of the user authentication. The protocol is composed of two stages: *Registration* and *Authentication*.

Registration
During registration a bilinear map, two hash functions and other system parameters are set. Secret keys are also exchanged between each user and cloud server. We give the definition of the bilinear map.

Definition 1 (Admissible bilinear map)**.** *Let G_1 and G_2 be two groups of order q for some large prime q. A map $\hat{e} : G_1 \times G_1 \to G_2$ is an admissible bilinear map if satisfies the following properties:*

1. Bilinear*: We say that a map $\hat{e} : G_1 \times G_1 \to G_2$ is bilinear if $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$ for all $P, Q \in G_1$ and all $a, b \in \mathbb{Z}$.*
2. Non-degenerate*: The map does not send all pairs in $G_1 \times G_1$ to the identity in G_2. Since G_1, G_2 are groups of prime order, if P is a generator of G_1, $e(P, P)$ is a generator of G_2.*
3. Computable*: There is an efficient algorithm to compute $\hat{e}(P, Q)$ for any $P, Q \in G_1$.*

Hash functions $H_1 : \{0;1\}^* \to G_1$, $H : \{0;1\}^* \to \mathbb{Z}_q^*$ and two random values $z, \gamma \in \mathbb{Z}_q^*$ are also chosen. In our identity-based setting let $Q = \gamma P$ denote the public and γ the secret key of the Private Key Generator (PKG). For each C_i a $(PK_i = Y_i = H_1(ID_i), SK_i = \gamma Y_i)$ keypair is generated, where the secret key is generated by PKG and ID_i is a publicly known identity information.

Between each U and C_i secret values D_i are exchanged securely in a way that only a digitally signed message is sent by U on a public channel. We assume the existence of a Bulletin Board ($\beta\beta$) that is publicly readable and can be written by PKG only. A public key exchange parameter $x_i P$ is stored for each C_i on $\beta\beta$. Cloud servers build a binary tree structure, secret key exchange parameter x_i is generated by the child elements, in case of leaf elements by the parent and the sister element. If a server breaks down, it can be replaced, therefore scalability and robustness of the cloud is assured.

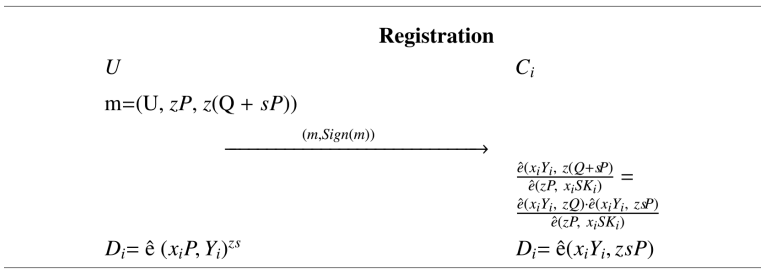

Figure 1: Registration phase

The calculations of the registration can be seen on Fig 1. Each user sends his password pw securely to C_{root}. The message is signed by U with an identity-based signature. C_{root} broadcasts $(m, Sign(m))$, each C_i calculates $D_i = \hat{e}(x_i Y_i, zsP)$. On the other side U computes and stores $\hat{e}(x_i P, Y_i)^z$, by calculating the sth power D_i is generated.

Authentication

In the authentication phase mutual authentication between the user and randomly chosen cloud servers is processed. U chooses k servers $(C_{i_1}, C_{i_2}, \ldots, C_{i_k})$ randomly. Cloud servers C_{i_j}, where $j = 1, \ldots, k$ verify the knowledge of the static password (pw) and the secret key (D_{i_j}) exchanged during Registration.

A random value v is generated by a time-based pseudo random number generator. Cloud server C_v performs the authentication. User U gives his password and his smart card computes $H(D_v^w) = H(\hat{e}(x_v P, Y_v)^{zsw})$. U chooses a bitstring w randomly and generates bitstrings $w_{i_1}, w_{i_2}, \ldots, w_{i_k}$ such that $w_{i_1} \oplus w_{i_2} \oplus \cdots \oplus w_{i_k} = w$, and each share is added to the hashed secret value exchanged during Registration. One can see that the random value w can be calculated on server side only if the cloud servers know the secret values exchanged. Therefore by generating the valid value $H(D_v^{w'}) \oplus w'$ authentication of cloud servers is completed.

Security analysis

Secure mutual authentication prevents adversaries to impersonate a legal user or a cloud server and to achieve illegal access to user data. In this section we give an informal security analysis of the proposed protocol. We consider vulnerability against the replay attack, impersonation attack, and different types of off-line attacks (e.g. dictionary attack, rainbow table attack).

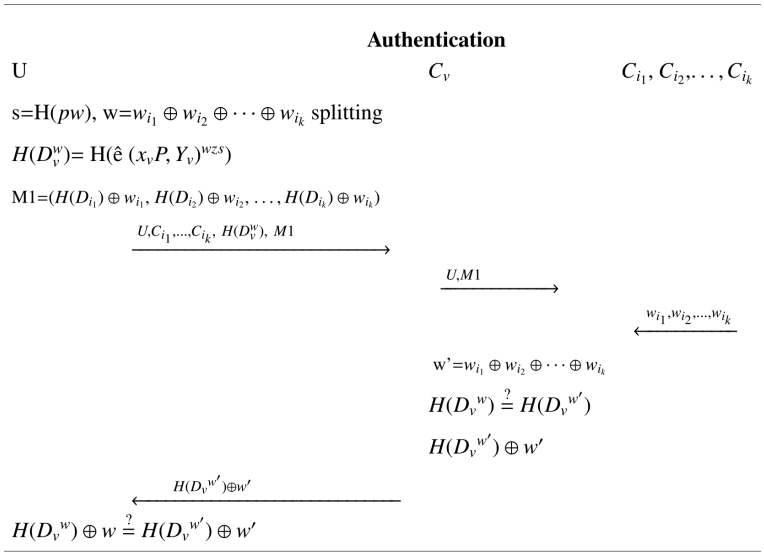

Figure 2: Authentication phase

A replay attack is successful, if by re-transmitting a previously sent message an adversary is able to authenticate himself. Since during authentication each message depends on a randomly generated w, therefore message freshness is achieved providing resistance against replay attack.

In order to impersonate a legal participant secret values D_i are necessary. Values D_i can be calculated only if both the password and secretly stored z, or both the cloud secret key and secret value x_i are known. Secret value x_i is not known by PKG, hence even PKG cannot carry out an impersonation attack. During authentication only the randomized hash values of D_i are sent. Due to the randomness and preimage, collision resistant hash functions no impersonation attack can be performed.

Most of the password-based authentication protocols use verification tables that are vulnerable against off-line attacks. A typical countermeasure is the use of salts. In our system the secret, random value z protects against week passwords.

Since the smart card does not store the password, the user always needs to give the password during the authentication phase, end-to-end identity verification is ensured.

Conclusion

We have designed an authentication protocol for the cloud environment which applies secret sharing and identity-based key exchange. Our system is robust against server breakdown and applies multiple-server identity verification. It is important to note that during the authentication phase we use $k+3$ hash calculations, $3k$ xor operations and two modular exponentiations, hence we have achieved an efficient scheme.

Acknowledgements

This work was supported by EFOP-3.6.2.-16-2017-00015. The project has been supported by the European Union, cofinanced by the European Scofial Fund. The first author is also supported by the Hungarian National Foundation for Scientic Research Grant No. NK 104208.

References

[1] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the Weil pairing. SIAM J. Comput., 32(3):586-615, 2003.

[2] J. Brainard, Ari Juels, Burt Kaliski, and Michael Szydlo, A New Two-Server Approach for Authentication with Short Secrets, Proceeding SSYM'03, Proceedings of the 12th conference on USENIX Security Symposium - Volume 12, pp 1-14, 2003.

[3] N. Chen, R. Jiang, Security Analysis and Improvement of User Authentication Framework for Cloud Computing, Journal of Networks, 9(1), pp. 198-203, 2014.

[4] A. J. Choudhury, P. Kumar, M. Sain, A Strong User Authentication Framework for Cloud Computing, Proceedings of IEEE Asia -Pacific Services Computing Conference, pp. 110-115, 2011.

[5] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. Journal of Computer and System Sciences, 28(2):270–299, 1984.

[6] A. Huszti, N. Olah, *A simple authentication scheme for clouds*, Proceedings of IEEE Conference on Communications and Network Security (CNS), (2016), Pages: 565 - 569.

[7] Hongwei Li, Yuanshun Dai, Ling Tian, and Haomiao Yang, Identity-Based Authentication for Cloud Computing, CloudCom 2009, LNCS 5931, pp. 157-166, 2009.

[8] M. S. Hwang, L. H. Li, A new remote user authentication scheme using smart cards, IEEE Transactions on Consumer Electronics, 46(1), pp. 28-30, 2000.

[9] W. C. Ku, S. M. Chen, Weaknesses and improvements of an efficient password based remote user authentication scheme using smart cards, IEEE Transactions on Consumer Electronics, 50(1), pp. 204-207, 2004.

[10] George Kurtz, Dmitri Alperovitch, Elia Zaitsev, *Hacking exposed: Beyond the Malware*, RSA 2015 (slide deck), https://www.rsaconference.com/ writable/presentations/ file_upload/expt10_hackingexposedbeyondthemalware.pdf, (2015).

[11] Ralph C. Merkle, *A Digital Signature Based on a Conventional Encryption Function*, Advances in Cryptology - CRYPTO '87, Lecture Notes in Computer Science, **293**, (1987), pp. 369-378.

[12] Miguel Soria-Machado, Didzis Abolins, Ciprian Boldea, Krzysztof Socha, *Kerberos Golden Ticket Protection, Mitigating Pass-the-Ticket on Active Directory*, CERT-EU Security Whitepaper 2014-007, (2016).